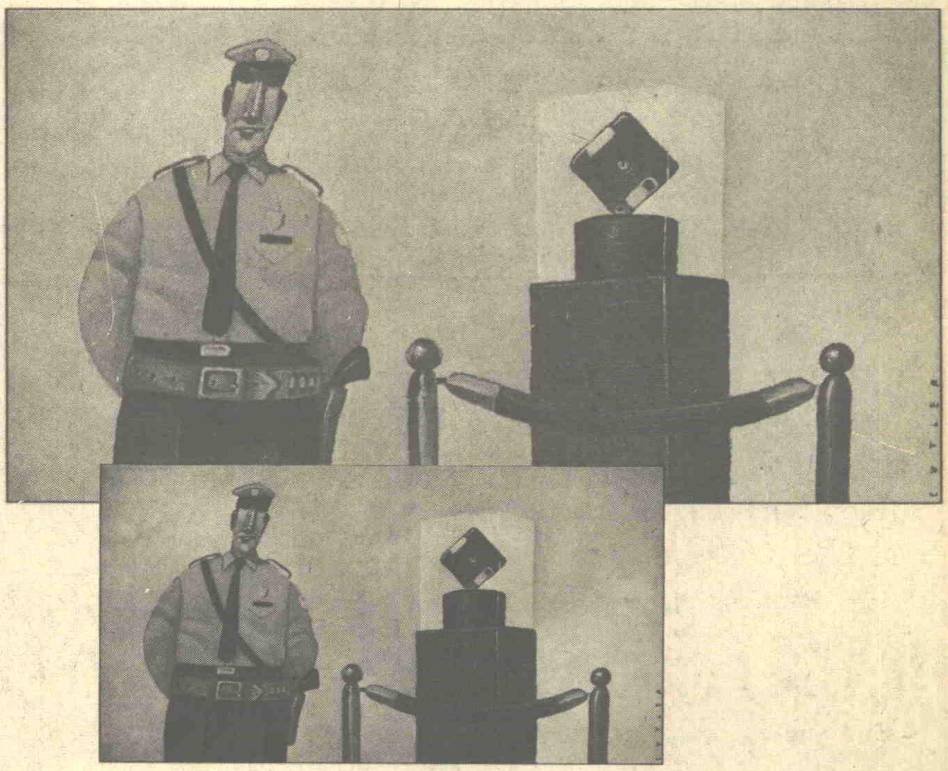
เทคนิคน่ารู้ 7 ประการใน VISUAL BASIC

เสนอเทคนิคอันแพรวพราว ที่ซ่อนอยู่ในวิชวลเบสิก เพราะอาจช่วยให้โปรแกรม ของท่านมีชีวิตชีวาขึ้น



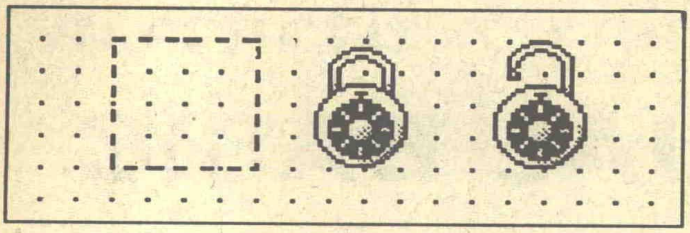
การเขียนโปรแกรมด้วยวิชวลเบสิกจะแตกต่างจากการเขียนโปรแกรมทั่ว ๆ ไป โดยมันจะประกอบไปด้วย 3 ขั้นตอนหลักคือ การออกแบบอินเตอร์เฟสของโปรแกรมด้วยทูลส์ต่าง ๆ การแก้ไขคุณสมบัติของอินเตอร์เฟสแต่ละอัน และการเขียนโปรแกรมควบคุมเหตุการณ์ที่จะเกิดขึ้นกับอินเตอร์เฟสเหล่านั้น สำหรับผู้ที่เริ่มเขียนโปรแกรมด้วยวิชวลเบสิกมาบ้างแล้วอาจรู้สึกว่าโปรแกรมของท่านค่อนข้างจะจืดชืด ไม่มีลูกเล่นอะไรที่จะดึงดูดผู้ใช้ได้เลย ถ้าท่านรู้สึกเช่นนั้นก็ไม่แปลก เป็นเพราะท่านยังไม่รู้ถึงเทคนิคอันแพรวพราวที่ซ่อนอยู่ภายในวิชวลเบสิก บทความต่อไปนี้จะนำเสนอเทคนิคบางประการที่อาจช่วยให้โปรแกรมของท่านดูมีชีวิตชีวาขึ้นมาได้

1. สร้างปุ่มกดที่เปลี่ยนไปตามสถานะของมัน

วิธีง่ายที่สุดที่จะสร้างเอฟเฟคท์ให้กับโปรแกรมของเราก็คือ การทำปุ่มกดที่เป็นรูปภาพให้มีการตอบสนองต่อผู้ใช้ เมื่อผู้ใช้คลิ๊กมัน มันก็จะเปลี่ยนรูปภาพไปตามสถานะของมัน ตัวอย่างเช่น เราสร้างปุ่มกดที่เป็นรูปภาพแสดงความหมายของสถานะที่ถูกล็อกและคลายล็อกของลูกกุญแจ เมื่อผู้ใช้คลิ๊กมัน มันก็จะเปลี่ยนสถานะจากล็อกเป็นคลายล็อก หรือกลับกัน

วิธีนี้เริ่มต้นด้วยการกำหนดรูป 2 รูปขึ้นมาในฟอร์ม แต่ละรูปจะแสดงถึงแต่ละสถานะของมัน และทั้ง 2 รูปนี้จะต้องมีคุณสมบัติ (property) ของ visible ไม่ถูกต้อง นั่นคือจะมองไม่เห็น จากนั้นจะกำหนดรูปอีก 1 รูปในฟอร์มนี้ ซึ่งจะเป็นรูปที่ใช้แสดงจริง ดังนั้นตำแหน่งของรูปนี้จะต้องอยู่ในที่ที่เราต้องการจะให้ปรากฏจริง และต้องมีคุณสมบัติของ visible เป็นความจริง เพียงแต่เรายังไม่ต้องกำหนดว่าจะให้เป็นรูปอะไร (แสดงด้วยเส้นประ) ขณะที่เรากำลังออกแบบโปรแกรมอยู่นี้เราจะเห็นรูปทั้ง 3 รูป แต่เวลารันจริงจะเห็นเพียงรูปเดียว คือรูปซ้ายมือ จอภาพขณะที่กำลังออกแบบโปรแกรมจะเป็นดังรูปที่ 1

ถ้าเริ่มรันโปรแกรมนี้มันจะทำโพรซีเดอร์ชื่อ Form_Load ก่อน ดังนั้นในโพรซีเดอร์นี้เราจะต้องกำหนดสถานะเริ่มต้นให้มันก่อน เช่น ตัวอย่างในรูปที่ 2 นี้จะกำหนดสถานะเริ่มต้น



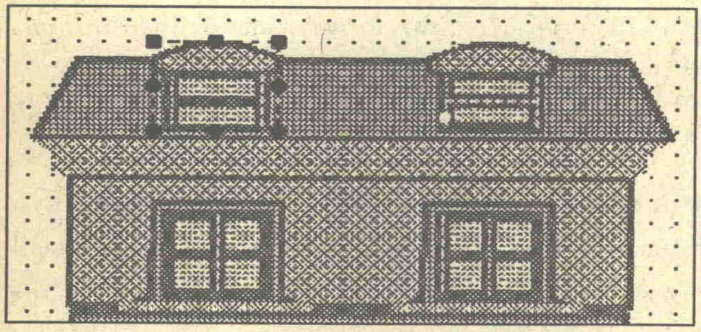
รูปที่ 1

```
Sub Form__Load < >
   PadLock.Picture = LockOpen.Picture
   LockFlag = False
   End Sub
```

รูปที่ 2

```
Sub PadLock__Click < >
  If LockFlag Then
     PadLock.Picture = LockOpen.Picture
  Else
     PadLock.Picture = LockClose.Picture
  End If
  LockFlag = Not LockFlag
End Sub
```

รูปที่ 3



รูปที่ 4

เป็นคลายล็อก โดยให้คุณสมบัติชื่อ picture ของรูปหนึ่งเท่ากับอีกรูปหนึ่ง

ในส่วนการตอบสนองของโปรแกรมเมื่อผู้ใช้คลิ๊กเม้าส์ที่รูปภาพ ก็จะทำการเปลี่ยนค่าแฟล็กที่ใช้เก็บสถานะเดิมของมันจากสถานะหนึ่งเป็นอีกสถานะหนึ่ง พร้อมกับนำรูปที่ 3 ซึ่งเป็นรูปที่ถูกต้องมาแสดง

ตัวอย่างของรูปต่าง ๆ ที่เราสามารถนำมาใช้ได้ถูกเก็บอยู่ในไดเรกทอรี \VB\ICONS\ ซึ่งถูกสร้างขึ้นมาตอนที่เราติดตั้งวิชวลเบสิก

2. ใช้ Hot spot กำหนดสิ่งที่เราสนใจ

ในการแสดงผลที่เป็นรูปภาพ เราอาจจะใช้ประโยชน์จากการสร้าง hot spot บนรูปภาพนั้น ๆ Hot spot ก็คือส่วนของพื้นที่สี่เหลี่ยมผืนผ้าที่อยู่ภายในรูปภาพ ซึ่งเราจะมองไม่เห็นมัน เมื่อผู้ใช้คลิ๊กเม้าส์หรือใช้เม้าส์ลากวัตถุมาทิ้งในบริเวณ hot spot เราอาจจะให้โปรแกรมทำอะไรเป็นพิเศษก็ได้ เช่น ให้แสดงคำอธิบายของพื้นที่ส่วนนั้น การสร้าง hot spot ทำได้ง่าย ๆ โดยเริ่มต้นจะกำหนดรูปภาพขนาดใหญ่กว่าพื้นที่ของ hot spot ขึ้นมาก่อน ซึ่งจะเป็นรูปที่ใช้แสดงนั่นเอง จากนั้นจึงกำหนด hot spot ขึ้นมาโดยการกำหนดรูปอีกรูปหนึ่งที่ยังไม่มีภาพบรรจุอยู่ (โดยให้คุณสมบัติชื่อ picture เท่ากับ none) รูปที่สองนี้จะเล็กกว่ารูปแรกและอยู่ทับรูปแรก ขอบเขตของรูปที่สองก็คือขอบเขตของ hot spot ในขณะรันโปรแกรมเราจะเห็นเพียงรูปแรกเท่านั้น ส่วนรูปที่สองจะมองไม่เห็น (เป็นรูปโปร่งใส)

ตัวอย่างในรูปที่ 4 นี้จะเป็นรูปภาพของบ้านหลังหนึ่งและมี hot spot อยู่ตรงบริเวณหน้าต่างรูปโดม

เมื่อเคอร์เซอร์ของเม้าส์อยู่เหนือบริเวณ hot spot เราอาจเปลี่ยนเคอร์เซอร์ของเม้าส์จากลูกศรให้เป็นกากบาทก็ได้ เพื่อให้ผู้ใช้ได้รู้ว่าเม้าส์ของเขาได้เข้ามาในบริเวณ hot spot แล้ว โดยการเปลี่ยนค่าของคุณสมบัติชื่อ Mouse-

Pointer ของรูปที่สองจากค่า 0 เป็น 2

หลังจากผู้ใช้คลิ๊กเม้าส์ในบริเวณ hot spot แล้ว ต้องการให้โปรแกรมตอบสนองผู้ใช้ด้วยการแสดงข้อความอธิบาย ก็สามารถทำได้โดยเพิ่มส่วนของโปรแกรมดังแสดงในรูปที่ 5

```
Sub Image5__Click < >
   MagBox "A| domer window", 24, "Architecture"
End Sub
```

รูปที่ 5

3. ตัวอักษร TAB ช่วยเพิ่มมิติของการแสดงผลใน list box

เนื่องจากลิสต์บ็อกซ์ในวิชวลเบสิกสนับสนุนการใช้ตัวอักษร TAB ทำให้การแสดงผลที่มีอักษร TAB สามารถจัดคอลัมน์ให้ตรงกันทุกบรรทัดได้ แต่ละบรรทัดจะมีตัวอักษร TAB อยู่ 2 ตัว ผลลัพธ์เป็นดังรูปที่ 6

Lawan Shawn	31	A
Somchai Saetung	44	N/A
Sunee Kemp	36	C
สมชาย แซ่ตั้ง	35	A
Sombat Cage	24	L/C

รูปที่ 6

เมื่อต้องการเพิ่มข้อความเข้าไปใน list box อาจเขียนโปรแกรมได้ตามรูปที่ 7

```
t = Chr$<9>
Customer__Name = "Somchai Saetung"
list1.AddItem pname + t + "44" + t + "N/A"
```

รูปที่ 7

เรายังสามารถให้โปรแกรมปรับความกว้างของลิสต์บ็อกซ์เพื่อไม่ให้แสดงผลข้อความในคอลัมน์ที่ 2 และ 3 ก็ได้ โดยการเปลี่ยนค่าคุณสมบัติชื่อ width ของลิสต์บ็อกซ์ให้เป็นค่าตามต้องการ ด้วยวิธีนี้เองทำให้การแสดงผลในลิสต์บ็อกซ์เปลี่ยนจากการแสดงผลในมิติเดียว (ตามแกน Y) มาเป็นการแสดงผลใน 2 มิติ (ตามแกน Y และ X)

4. ป้องกันการโหลดโปรแกรมซ้ำซ้อน

มีบ่อยครั้งที่เราไม่ต้องการให้โปรแกรมของเราถูกเรียกมารันครั้งที่สอง โดยที่การรันครั้งแรกยังไม่จบโปรแกรมเลย หรือจะกล่าวว่า เราต้องการให้โปรแกรมของเรามีอินสแตนซ์เกิดขึ้นในเวลาใดเวลาหนึ่งเพียงอินสแตนซ์เดียวเท่านั้น เพราะบางครั้งโปรแกรมของเราอาจต้องเขียนข้อมูลลงไฟล์หรือต้องสร้างไฟล์ชั่วคราวขึ้นมา ถ้ามีอินสแตนซ์ของโปรแกรมแอคทีฟอยู่หลายอินสแตนซ์ อาจทำให้บางอินสแตนซ์ทำงานไม่ถูกต้องก็ได้ เช่น อินสแตนซ์หนึ่งกำลังลบไฟล์ทิ้ง ส่วนอีกอินสแตนซ์หนึ่งพยายามอ่านไฟล์ที่ถูกลบ เป็นต้น

ในวิชวลเบสิกได้เตรียมการไว้ให้เรียบร้อยแล้ว โดยวิชวลเบสิกจะกำหนดให้โปรแกรมของเราเป็นออปเจ็กต์อันหนึ่งที่มีชื่อว่า App (ย่อมาจาก Application) และออปเจ็กต์นี้มีคุณสมบัติอันหนึ่งชื่อ PrevInstance ที่เก็บค่า Boolean เราสามารถนำค่านี้มาเช็คว่ามีอินสแตนซ์อื่นของโปรแกรมนี้กำลังรันอยู่หรือเปล่า ถ้ามีจะทำให้ PrevInstance เป็นจริง มิฉะนั้นจะเป็นเท็จ

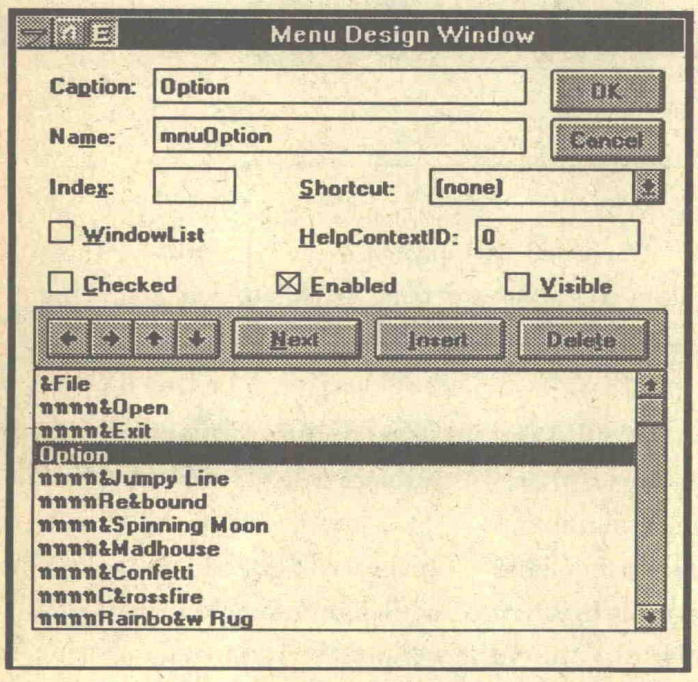
โปรแกรมในรูปที่ 8 นี้ เป็นตัวอย่างโปรแกรมที่ใช้เช็คว่าโปรแกรมนี้แอคทีฟอยู่เพียงอินสแตนซ์เดียวหรือไม่ ถ้าไม่ใช่จะมีเมสเสจแจ้งเตือน แล้วจบโปรแกรม (อินสแตนซ์) นี้ ให้สังเกตว่า EXEName เป็นอีกคุณสมบัติหนึ่งของ App ที่เก็บชื่อโปรแกรมของเรา

5. ป๊อปอัพเมนูน่าใช้กว่าเมนูธรรมดาเป็นไหน ๆ

จุดเด่นอย่างหนึ่งของวิชวลเบสิกเวอร์ชัน 3.0 ก็คือ ป๊อปอัพเมนู เราสามารถสร้างป๊อปอัพเมนูได้โดยเริ่มต้นสร้างเมนูธรรมดา โดยกด Ctrl-M จะปรากฏวินโดวส์ดังรูปที่ 9 หลังจากสร้างเมนูตามปกติแล้ว จากนั้นใช้เซตคุณสมบัติชื่อ visible ของเมนูอันที่

```
Sub Form_Load < >
   If App.PrevInstance Then
      msg$ = App.EXEName & "already running"
      MsgBox msg$, 48
      End
   End If
End Sub
```

รูปที่ 8



รูปที่ 9

```
Sub Form_MouseUp <Button As Integer,
   If Button = 2 Then
      PopupMenu mnuOption
   End If
End Sub
```

รูปที่ 10

รูปที่ 11

รูปที่ 12

ต้องการให้เป็นป๊อปอัพเมนูให้มีค่าเป็น false เพื่อไม่ให้เมนูนี้ปรากฏที่เมนูบาร์ ก็จะเสร็จสิ้นขั้นตอนการสร้างเมนู

ต่อไปก็คือการทำให้ป๊อปอัพเมนูของเราปรากฏขึ้นที่จอภาพ โดยใช้คำสั่ง PopupMenu และตามด้วยชื่อเมนูที่ต้องการ ในตัวอย่างในรูปที่ 10 นี้เราจะใช้เมนูชื่อ mnuOption มันจะแอคตีฟเมื่อผู้ใช้คลิ๊กเม้าส์ปุ่มขวาบนฟอร์มนั้น จะทำให้มีเมนูเกิดขึ้นดังรูปที่ 11

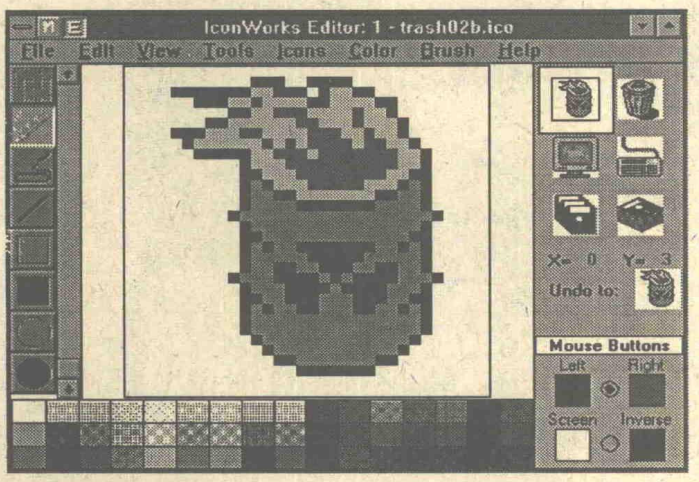
6. เมื่อข้อความในเท็กซ์บ็อกซ์ถูกมาร์คอัตโนมัติทุกครั้งที่เราเริ่มใช้มัน

ทุกครั้งที่เราใช้เท็กซ์บ็อกซ์ในการรับอินพุตจากผู้ใช้ มันจะแสดงค่าเดิมของมัน ถ้าเราต้องการใส่ค่าใหม่ เราจำเป็นจะต้องลบค่าเดิมทิ้งก่อน แล้วจึงจะคีย์ค่าใหม่ได้ ทำให้ไม่สะดวกอย่างยิ่ง ถ้าเราสังเกตแอพพลิเคชันอื่น ๆ บนวินโดวส์จะพบว่า ทุกครั้งที่เราใช้เท็กซ์-บ็อกซ์ในการรับค่า มันจะทำการมาร์คข้อความเดิมให้เอง ทำให้เราเห็นข้อความแบบกลับขาว-ดำ ดังรูปที่ 12 ถ้าเราต้องการใส่ค่าใหม่ก็สามารถคีย์เข้าไปได้เลย โดยข้อความเดิมจะถูกลบทิ้งไปโดยอัตโนมัติ

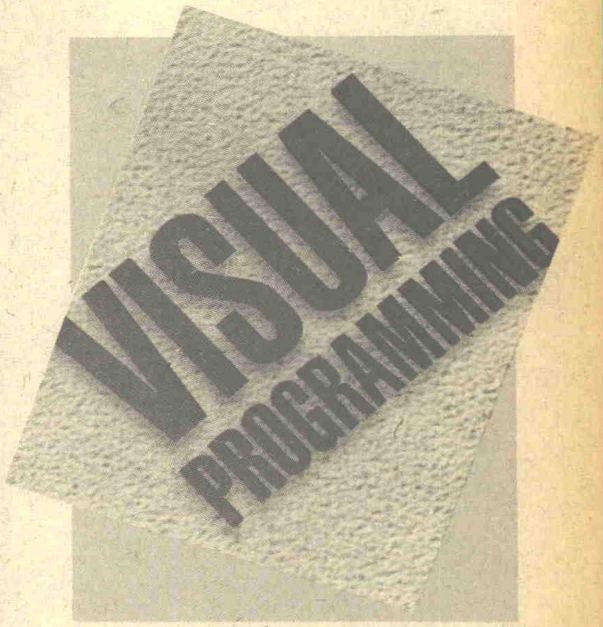
เราสามารถทำให้เท็กซ์บ็อกซ์ของเราเป็นอย่างนั้นได้โดยไม่ยาก ด้วยการ

```
Sub Text1__GotFocus < >
| text1.SelStart = 0
  text1.SelLength = Len <text1.Text>
End Sub
```

รูปที่ 13



รูปที่ 14



เพิ่มโปรแกรมอีก 2 บรรทัดในโพรซีเดอร์ GotFocus ดังรูปที่ 13

ในโพรซีเดอร์นี้จะกำหนดให้ SelStart ชี้ที่ตำแหน่งแรกของข้อความ และให้ SelLength เป็นความยาวของข้อความที่ถูกเลือก ซึ่งหาได้จากฟังก์ชัน Len

7. ไอคอนก็เป็นเอกลักษณ์อย่างหนึ่งของโปรแกรม

หลังจากที่เราออกแบบโปรแกรมของเราจนเสร็จสิ้นแล้ว ขั้นตอนสุดท้ายที่ลืมเสียไม่ได้ก็คือ การกำหนดรูปไอคอนให้กับโปรแกรมของเรา รูปไอคอนนี้ใช้สำหรับแสดงอยู่ในโปรแกรมแมนเนเจอร์ และยังใช้แสดงตอนที่เราเลือก Minimize โปรแกรมให้เป็นไอคอน ดังนั้นไอคอนจึงเปรียบเสมือนเป็นหน้าตาของโปรแกรมเราเชียวล่ะ

การกำหนดรูปไอคอนให้กับโปรแกรมของเรา ทำได้โดยกำหนดไฟล์ไอคอน (ไฟล์นามสกุล .ICO) ให้กับคุณสมบัติชื่อ icon ของฟอร์มนั้น ตัวอย่างไฟล์ไอคอนที่ติดมากับวิชวลเบสิกจะอยู่ในไดเรกทอรี \VB\ICONS แต่ถ้าเราต้องการสร้างไอคอนที่เป็นของเราเอง ก็ย่อมทำได้ โดยใช้โปรแกรมพวกไอคอนเอดิเตอร์

ในวิชวลเบสิกมีโปรแกรมชื่อ ICONWORKS.MAK อยู่ในไดเรกทอรี \VB\SAMPLES\ICONWORKS ซึ่งเป็นโปรแกรมตัวอย่างที่สามารถใช้งานได้จริง ๆ โปรแกรมนี้ใช้สำหรับแสดง แก้ไข หรือสร้างไอคอน มันสามารถแสดงรูปไอคอนทั้งหมดจากไฟล์ไอคอนต่าง ๆ ในไดเรกทอรีที่กำหนดบนจอภาพ เราสามารถเลือกรูปมาแก้ไขหรือสร้างใหม่ได้ตามต้องการ หน้าตาของโปรแกรมเป็นดังรูปที่ 14

การใช้งานโปรแกรม ICONWORKS นี้จะต้องโหลดโปรแกรมนี้เข้ามาในวิชวลเบสิกก่อน แล้วสั่งรันได้เลย หรือถ้าจะทำเป็นเอ็กซีคิวต์ไฟล์เพื่อจะได้เรียกใช้ได้จากโปรแกรมแมนเนเจอร์ ก็ให้เลือกหัวข้อ Make EXE จากเมนู File นอกจากโปรแกรมนี้จะใช้งานได้ดีแล้ว ยังเหมาะที่จะใช้ศึกษาการเขียนโปรแกรมด้วยวิชวลเบสิกอย่างยิ่ง